

满分70 前15 2分一个，判断题1.5，剩下3分一个

1.单选题 (1.5分)

windows操作系统中下列哪个是复制操作()

A.control + C B.control + V C.control + P D.control + A

2.单选题 (2分)

5个小朋友站成一个环，其中有两个双胞胎，如果要求这两个双胞胎必须相邻，则有（ ）种不同排列方法。

A. 12

B. 36

C. 24

D. 48

3.单选题 (2分)

若两条直线相交则会有一个交点，那么五条直线相交，最多能够有（ ）个交点。

A.5

B.10

C.12

D.15

4.单选题 (2分)

由3个a，5个b和2个c构成的所有字符串中，包含子串“abc”的共有（ ）个。

A. 40320 B. 39600 C. 840 D. 780

5.单选题 (2分)

十进制-15的原码反码补码分别是()

A.(11111)原 (10000)反 (10001)补 B.(01111)原 (01111)反 (01111)补 C.(11111)原 (00000)反 (00001)补 D.(11111)原 (10000)反 (11111)补

6.单选题 (2分)

```
int a[10][5] = {0};
printf("%d %d %d %d %d",a,a + 2,a[0] + 1,&a[2] + 3,a[7][0] + 1);
```

假设 a的地址是 0，那么答案的预期值是多少（ ）？

A. 0 40 4 52 144 B. 0 8 4 52 144 C. 0 40 4 100 1 D. 0 8 20 100 1

7.单选题 (2分)

-11 >> 1， 11 >> 3， 10 << 2， -10 << 1 的值分别是多少()？

A.-5,1,40,-20 B.-6,2,20,-10 C.-6,2,40,-10 D.-5,1,20,-20

8.单选题 (2分)

-11 & -7, -5 | -11, 17^{-11} 的值分别是多少()?

A.-3,-1,-7 B.15,-1,-28 C.-15,-9,-7 D.-15,-1,-28

9.单选题 (2分)

```
int dg(int x){
    if(x < 1) return 1;
    return 2 * dg(x / 2);
}
```

请问 dg(31) 最后的答案是多少()

A.30 B.16 C.32 D.64

10.单选题 (2分)

```
int dg(int x){
    if(x < 1) return 1;
    return 2 * dg(x / 2);
}
```

请问这段代码的复杂度是多少()

A. $O(x)$ B. $O(1)$ C. $O(\log(x))$ D. $O(2^x)$

11.单选题(2分)

```
void selection_sort(int n) {
    for (int i = 1; i < n; ++i) {
        int ith = i;
        for (int j = i + 1; j <= n; ++j) {
            if (a[j] < a[ith]) {
                ith = j;
            }
        }
        std::swap(a[i], a[ith]); //交换, algorithm工具库里的
    }
}
```

请问以上代码是什么排序()?

A.插入排序 B.选择排序 C.归并排序 D.冒泡排序

12.单选题(2分)

请问插入排序最好的复杂度, 最坏的复杂度是多少()

A. $O(n)$ $O(n)$ B. $O(n^2)$ $O(n^2)$ C. $O(n)$ $O(n^2)$ D. $O(n\log n)$ $O(n^2)$

13.单选题(2分)

请问冒泡排序最好的复杂度, 最坏的复杂度是多少()

A. $O(n)$ $O(n)$ B. $O(n^2)$ $O(n^2)$ C. $O(n)$ $O(n^2)$ D. $O(n\log n)$ $O(n^2)$

14.单选题(2分)

请问选择排序最好的复杂度，最坏的复杂度是多少()

A. $O(n)$ $O(n)$ B. $O(n^2)$ $O(n^2)$ C. $O(n)$ $O(n^2)$ D. $O(n\log n)$ $O(n^2)$

15.单选题(2分)

请问以下关于递归的说法，哪个是正确的()

A.递归是不断地调用自身的编程技巧 B.递归和循环一样 C.递归是贪心的思想 D.递归会将问题复杂化

16.填空题(12分)

```
1  #include <stdio>

2  #include <cstring>

3  using namespace std;

4  char st[100];

5  int main() {

6      scanf("%s", st);

7      int n = strlen(st);

8      for (int i = 1; i <= n; ++i) {

9          if (n % i == 0) {

10             char c = st[i - 1];

11             if (c >= 'a')

12                 st[i - 1] = c - 'a' + 'A';

13         }

14     }

15     printf("%s", st);

16     return 0;

17 }
```

•判断题 (1.5分一个)

- 1) 输入的字符串只能由小写字母或大写字母组成。 ()
- 2) 若将第8行的“i = 1”改为“i = 0”，程序运行时会发生错误。 ()
- 3) 若将第8行的“i <= n”改为“i * i <= n”，程序运行结果不会改变。 ()
- 4) 若输入的字符串全部由大写字母组成，那么输出的字符串就跟输入的字符串一样。 ()

•选择题 (2分一个)

5) 若输入的字符串长度为18,那么输入的字符串跟输出的字符串相比,至多有 () 个字符不同。

6) 若输入的字符串长度为 () , 那么输入的字符串跟输出的字符串相比,至多有36个字符不同。

1.

A. 正确

B. 错误

2.

A. 正确

B. 错误

3.

A. 正确

B. 错误

4.

A. 正确

B. 错误

5.

A. 18

B. 6

C. 10

D. 1

6.

A. 36

B. 100000

C. 1

D. 128

17.填空题 (13.5分)

```
1  #include<stdio.h>
2  int sum = 0;
3  void dg(int x,int y){
4      if(y == 1){
5          sum++;
6          return;
7      }
8      for(int i = x; i <= y; i++){
9          if(y % i == 0){
10             dg(i,y / i);
11         }
12     }
13     return;
14 }
15 int main(){
```

```

16     int n;
17     scanf("%d",&n);
18     for(int i = 1; i <= n; i++){
19         int x;
20         scanf("%d",&x);
21         dg(2,x);
22         printf("%d\n",sum);
23         sum = 0;
24     }
25     return 0;
26 }

```

假设输入的 n 是不超过 2^{63} 的正整数， k 都是不超过 10000 的正整数，完成下面的判断题和单选题：

•判断题

- 1) 若 $n = 1$, $x = 11$ 的时候, 输出 1
- 2) 若 $n = 1$, $x = 12$ 的时候, 输出 3
- 3) 若 把第 23 行删掉, 答案不会改变

•单选题

- 4) 若输入 2 3 8 则输出()。
- 5) 若第21行改成 $dg(x,x)$, 输入 3 12 20 5 会输出()
- 6) 若输入的 1 6 函数最多被调用几次()

1.

- A. 正确
- B. 错误

2.

- A. 正确
- B. 错误

3.

- A. 正确
- B. 错误

4.

- A. 1 3
- B. 1 4
- C. 2 3
- D. 2 4

5.

- A. 4 4 1
- B. 4 1 4
- C. 1 1 1

D.1 4 4

6.

A. 5

B. 4

C. 3

D. 6

18.程序设计题

输入长度为 n 的一个正整数序列，要求输出序列中最长连号的长度。

连号指在序列中，从小到大的连续自然数。

输入格式

第一行，一个整数 n 。

第二行， n 个整数 a_i ，之间用空格隔开。

输出格式

一个数，最长连号的个数。

输入输出样例

输入 #1

复制

输出 #1

复制

```
10
1 5 6 2 3 4 5 6 8 9
```

```
5
```

试补全程序。

```
#include<stdio.h>

int b[11000];

int main(){

    int n;
    scanf("%d",&n);
    int s = 0;
    int maxn = 0;
    for(int i = 1; i <= n; i++){
        scanf("%d",&b[i]);
        if( <1> ){
            s = s + 1;
        }else{
            <3>
        }
        if( <2> ){
            maxn = s;
        }
    }

    printf("%d",maxn);
```

```
    return 0;
}
```

1. <1> 这个空填什么()
A. $b[i] = b[i - 1] + 1$ B. $b[i] = b[i + 1] + 1$ C. $b[i - 1] = b[i] + 1$ D. $b[i + 1] = b[i] + 1$
2. <2> 这个空填什么()
A. $\max n > s$ B. $\max n < s$ C. $\max n > b[i]$ D. $\max n < b[i]$
3. <3> 这个空填什么()
A. $s = s + 1$ B. $s = s - 1$ C. $s = s * 2$ D. $s = 1$

19.程序设计题

题目描述

在 $n \times n$ 的格子中有 m 个地毯。

给出这些地毯的信息，问每个点被多少个地毯覆盖。

输入格式

第一行，两个正整数 n, m 。意义如题所述。

接下来 m 行，每行两个坐标 (x_1, y_1) 和 (x_2, y_2) ，代表一块地毯，左上角是 (x_1, y_1) ，右下角是 (x_2, y_2) 。

输出格式

输出 n 行，每行 n 个正整数。

第 i 行第 j 列的正整数表示 (i, j) 这个格子被多少个地毯覆盖。

输入输出样例

输入 #1

```
5 3
2 2 3 3
3 3 5 5
1 2 1 4
```

复制

输出 #1

复制

```
0 1 1 1 0
0 1 1 0 0
0 1 2 1 1
0 0 1 1 1
0 0 1 1 1
```

试补全程序

思路: 对于每一行做一个差分数组

```
#include<iostream>
using namespace std;
long long n,a[1100][1100],b[1100][1100];
int main() {

    int n,m;
    scanf("%d %d",&n,&m);
    for(int i = 1; i <= m; i++){
        int x1,y1,x2,y2;
        scanf("%d %d %d %d",&x1,&y1,&x2,&y2);
        for(int x = x1; x <= x2; x++){
            <1>
            b[x][y1] += 1;
        }
    }
}
```

```

    for(int i = 1; i <= n; i++){
        for(int j = 1; j <= n; j++){
            <2>
        }
    }

    for(int i = 1; i <= n; i++){
        for(int j = 1; j <= n; j++){
            printf("%d ",a[i][j]);
        }
        printf("\n");
    }
}

```

1. <1>这个空填什么

A. $b[x][y2 + 1] - = 1;$ B. $b[x][y2 + 1] + = 1;$ C. $b[x - 1][y2 + 1] + = 1;$ D. $b[x - 1][y2 + 1] - = 1;$

2. <2>这个空填什么

A. $a[i][j] = a[i][j - 1] + b[i][j];$ B. $a[i][j] = a[i][j - 1] - b[i][j];$ C. $a[i + 1][j + 1] = a[i][j] + b[i][j];$ D. $a[i][j + 1] = a[i][j] + b[i][j];$